

Sunmi-K1 打印服务开发文档

Sunmi-K1 打印服务开发文档	1
简介:	3
AIDL方式连接打印服务:	4
伪代码示例:	4
打印服务接口说明:	5
1、打印机初始化	5
2、获取打印机状态	5
3、打印机按行高走纸	5
4、打印机按像素走纸	6
5、刷新缓冲区	6
6、水平定位	6
7、打印文本内容	6
8、打印条形码	7
9、打印二维码	7
10、打印图片	7
11、设置横向水平定位点	8
12、设置对齐方式	8
13、切刀切纸	8
14、发送数据指令	8
15、设置字体大小	9
16、表格打印	9
接口返回值说明:	9
条码类型说明:	9
状态反馈:	10

简介：

本文档主要用于介绍商米开发者如何在Sunmi-K1机器上使用打印服务，
实现自己的打印任务；

Sunmi-K1通过Aidl调用打印机；

AIDL方式连接打印服务：

AIDL 是 Android Interface Definition language 的缩写，它是一种 Android 内部进程通信接口的描述语言，通过它我们可以定义进程间的通信接口。商米 AIDL 提供封装好的常用打印指令，方便开发者快速接入 Sunmi 打印机，同时 Sunmi 也支持所有常用的《ESC/POS》指令集，可以通过接口直接发送指令集控制打印机。

建立连接可分以下 5 步骤：

1. 在项目中添加资源文件中附带的 AIDL 文件（部分机型还包含 java 文件）。
2. 在控制打印的代码类中实现 ServiceConnection。
3. 调用 `ApplicationContext.bindService()`，并在 `ServiceConnection` 实现中进行传递。注意：`bindService` 是非阻塞调用，意味着调用完成后并没有立即绑定成功，必须以 `serviceConnected` 为准。
4. 在 `ServiceConnection.onServiceConnected()` 实现中，你会接收一个 `IBinder` 实例(被调用的 Service)。调用 `ExtPrinterService.Stub.asInterface(service)` 将参数转换为 Aidl 文件中对应的打印服务类类型。
5. 现在就可以调用 Aidl 接口中定义的各种方法进行打印了。

伪代码示例：

绑定服务

```
Intent intent = new Intent();  
intent.setPackage("com.sunmi.extprinterservice");  
intent.setAction("com.sunmi.extprinterservice.PrinterService");  
bindService(intent, serviceConnection, Context.BIND_AUTO_CREATE);
```

需要新建一个 `ServiceConnection` 服务绑定回调

```
ServiceConnection serviceConnection = new ServiceConnection() {  
    @Override  
    public void onServiceConnected(ComponentName name, IBinder service) {  
        ExtPrinterService interface = ExtPrinterService.Stub.asInterface(service);  
    }  
  
    @Override  
    public void onServiceDisconnected(ComponentName name) {  
  
    }  
};
```

使用 `interface` 对象实现自己的打印任务

```
interface.printText("123456\n");
```

使用结束后解绑服务

```
unbindService(serviceConnection);
```

打印服务接口说明：

1、打印机初始化

方法	int printerInit()
说明	对打印机初始化，将会清空打印缓冲区的数据，但不影响接收缓冲区，并把之前所有设置的属性恢复为默认状态
输入	无
返回	见 接口返回值

2、获取打印机状态

方法	int getPrinterStatus()
说明	获取打印机运行状态
输入	无
返回	-1 打印机脱机或打印服务还未连接打印机 0 打印机运行正常 1 打印机开盖 2 打印机缺纸 3 打印机即将缺纸 4 打印机过热

3、打印机按行高走纸

方法	Int lineWrap(int n)
说明	打印机走纸n行 如果打印机打印缓冲区有数据，将输出数据并走纸n行 如果行高被设置为0，则走纸距离为0
输入	行数 (0<n<256)
返回	见 接口返回值

4、打印机按像素走纸

方法	Int pixelWrap(int n)
说明	打印机走纸n点行 如果打印机打印缓冲区有数据，将输出数据并走纸n点行
输入	像素点数 (0<n<256)
返回	见 接口返回值

5、刷新缓冲区

方法	int flush()
说明	刷新打印缓冲区，当缓冲区有数据时将输出，如果没有数据将进纸一行
输入	无
返回	见 接口返回值

6、水平定位

方法	int tab()
说明	将打印位置移动到下一个水平定位点位置 如果水平定位点超出打印区域则移动到行尾 如果此时已经在行尾则会执行换行操作
输入	无
返回	见 接口返回值

7、打印文本内容

方法	int printText(String text)
说明	此方法会将输入文本转为对应字符集编码的十六进制字节流 打印服务默认将文本内容转换为gb18030编码
输入	text要打印的文本内容
返回	见 接口返回值

8、打印条形码

方法	int printBarCode(String code, int type, int width, int height, int hriPos)
说明	打印自定内容的条形码
输入	code 条形码内容（根据条码类型不同，条码内容需满足其格式） type 条形码类型 0: UPC-A 1: UPC-E 2: EAN13 3: EAN8 4:CODE39 5:ITF 6:CODABAR 7:CODE93 8:CODE128 见 条码类型备注 width 条形码宽度 2-6 像素（若条码宽度设置过宽，整个条码超过纸张宽度将不会输出条码内容） height 条形码高度 1-255 像素 hriPos HRI位置 0: 不打印 1: 条形码上方 2: 条形码下方 3: 条形码上下方
返回	见 接口返回值

9、打印二维码

方法	int printQrCode(String code, int modeSize, int errorlevel)
说明	打印二维码
输入	code 要打印的二维码内容，默认为utf-8字符集 modeSize 二维码块大小 1-16 像素点 errorlevel 二维码纠错等级 0-3 四个等级
返回	见 接口返回值

10、打印图片

方法	int printBitmap(Bitmap bitmap, int mode)
说明	将bitmap图转成光栅位图方式打印 此方法适合打印宽度在打印纸内的图片 数据过大时串口流控模式会阻塞，无流控时会丢数据
输入	bitmap 要打印的bitmap图像，图像宽高均为8的整数倍 mode 0: 普通 1: 倍宽 2: 倍高 3: 倍高倍宽
返回	见 接口返回值

11、设置横向水平定位点

方法	int setHorizontalTab(int[] k)
说明	将标记水平定位的位置，每个标记点位置由k[n]个ascii字符宽度指定，默认的水平定位点以8个ascii字符宽度为一个定位点
输入	k 横向定位点数组k[n]， 数组长度n最大可以是32， 0 <k[n]<256 数组中必须是升序排列， 否则将抛出异常 k设置为null时将恢复默认定位点， 默认位置为8个字符间隔
返回	见 接口返回值

12、设置对齐方式

方法	int setAlignMode(int type)
说明	设置打印内容的对齐方式
输入	0居左（默认）、1居中、2居右
返回	见 接口返回值

13、切刀切纸

方法	int cutPaper(int m, int n)
说明	切纸
输入	m 切纸模式 0: 全切 1: 半切 2: 进纸切纸 n 进纸距离 此参数只有在设置 m=2时有效， 由于打印机型号不同切刀到打印头距离不同， 当n=0时将自动走纸此距离， n>0将走额外设置距离
返回	见 接口返回值

14、发送数据指令

方法	int sendRawData(byte[] cmd)
说明	发送epson控制指令
输入	cmd 发送的epson指令
返回	见 接口返回值

15、设置字体大小

方法	int setFontZoom(int hori, int veri)
说明	由于打印机硬字库限制，字体大小仅能倍数放大，本方法可以控制字体在横向和纵向方向的放大倍数
输入	hori、veri 的范围为1-8，表示在横向、纵向上字体的放大倍数，若设置在范围外将返回错误参数
返回	见 接口返回值

16、表格打印

方法	int printColumnsText(String[] colsTextArr, int[] colsWidthArr, int[] colsAlign)
说明	以表格方式输出打印内容，每个数组表示在此列上的数据及格式，需要多次调用此方法才可以达到表格输出的样式效果 ⚠注意： 调用此方法后会将之前设置样式初始化！
输入	colsTextArr: 每列要打印的内容，支持中文及ascii码 colsWidthArr: 每列可容纳的最大字符数量， 字符数以ascii码个数为单位（一个中文等于两个ascii码数量） ，当文本内容超出可容纳的最大数量时将移动到本列下一行，当所有列的最大字符数量超出一行所能容纳的字符数时，将不会打印 colsAlign: 每列内容的对齐方式，仅当内容字符数小于最大字符数量时才有效果
返回	见 接口返回值

接口返回值说明：

由于打印机执行打印任务与发送数据是异步执行，所以接口返回并不代表实际打印是否成功，接口返回值除特殊说明外，均表示此次指令由打印服务接收的情况，因此可能打印机处于异常状态如缺纸等仍可接收打印数据，当异常恢复后继续执行缓存打印数据；

当返回值 ≥ 0 时表示此命令发送成功，打印机将处理此指令；

成功返回的具体值参考打印机状态返回值；

当返回值 < 0 时表示此次命令发送失败，不会执行打印任务，具体错误如下：

- 1 打印机离线或打印机未准备好
- 2 缓存已满，不接收打印数据
- 3 发送数据异常
- 4 发送命令或参数错误

条码类型说明：

code39 最长打印 13 个数字

code93 最长打印 17 个数字

code128 code128 分三类：{A},{B},{C；A 类：包大写字母、数字、标点等，B 类：大小写字母，数字；C 类：纯数字；默认 B 类编码，若要使本编码需在内容前面加“{A”、“{B”、“{C”声明使用的类型，可以混合使用，例如：“{A2344A”，”{C123123”，”{A1A{B13Bxc{C12”。

ean8 要求 8 位数字（最后一位校验位），有效长度 8 个数字

ean13 有效长度 13 个数字，其中最后一位为校验位

ITF 要求输入数字，且有效小于 14 位，必须是偶数位

状态反馈：

打印机的状态可通过调用接口 `getPrinterStatus ()` 方法主动获取；

打印机的状态还可以通过注册广播异步获取：

功能	action
可以打印	com.sunmi.extprinterservice.NORMAL_ACTION
打印机离线	com.sunmi.extprinterservice.OFFLINE_ACTION
打印机仓盖未关闭	com.sunmi.extprinterservice.COVER_OPEN_ACTION
打印机缺纸	com.sunmi.extprinterservice.OUT_OF_PAPER_ACTION
打印机纸将尽	com.sunmi.extprinterservice.LESS_OF_PAPER_ACTION
打印机过热	com.sunmi.extprinterservice.HOT_ACTION